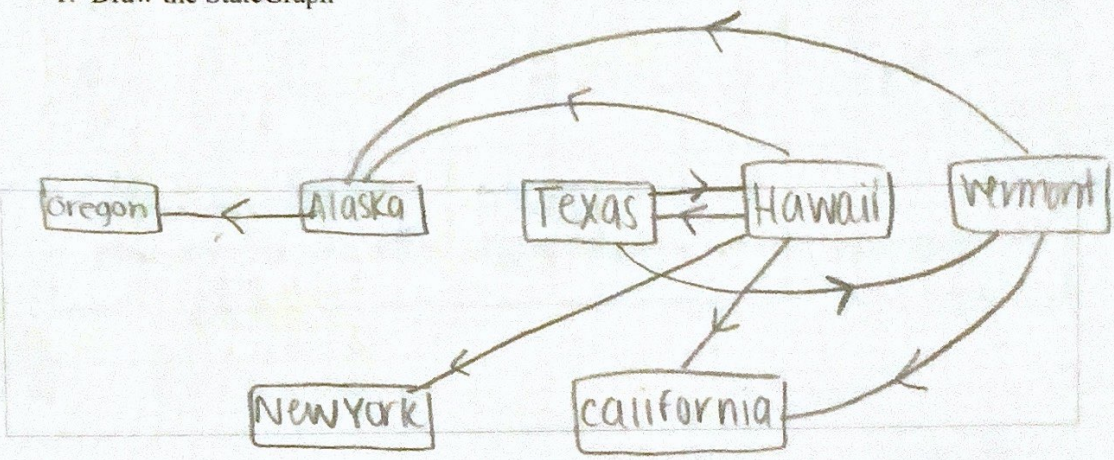
Odalis Flores

CMSC204
Kartchner

V(StateGraph) = {Oregon, Alaska, Texas, Hawaii, Vermont, NewYork, California}
E(StateGraph) = {(Alaska, Oregon), (Hawaii, Alaska), (Hawaii, Texas), (Texas, Hawaii), (Hawaii, California), (Hawaii, New York), (Texas, Vermont), (Vermont, California), (Vermont, Alaska)}

1. Draw the StateGraph



1. Describe the graph pictured above, using the formal graph notation.

V(StateGraph) = 7 {Oregon, Alaska, Texas, Hawaii, Vermont, NewYork, California}

E(StateGraph) = 9 {(Alaska, Oregon), (Hawaii, Alaska), (Hawaii, Texas), (Texas, Hawaii), (Hawaii, California), (Hawaii, New York), (Texas, Vermont), (Vermont, California), (Vermont, Alaska)}

2. a. Is there a path from Oregon to any other state in the graph?
No path from oregon to another state

b. Is there a path from Hawaii to every other state in the graph?
yes

c. From which state(s) in the graph is there a path to Hawaii?
Texas

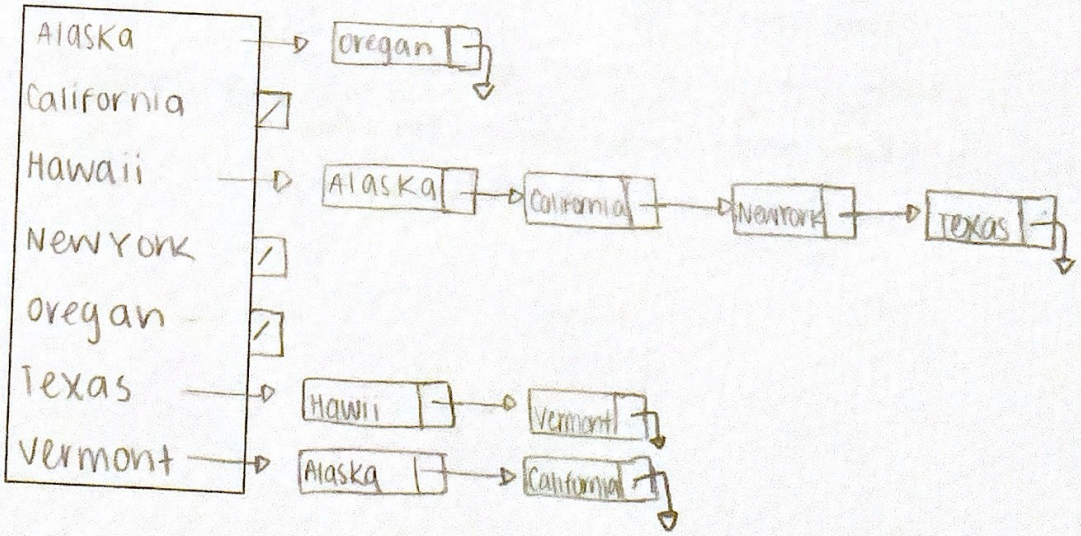
3. a. Show the adjacency matrix that would describe the edges in the graph. Store the vertices in alphabetical order

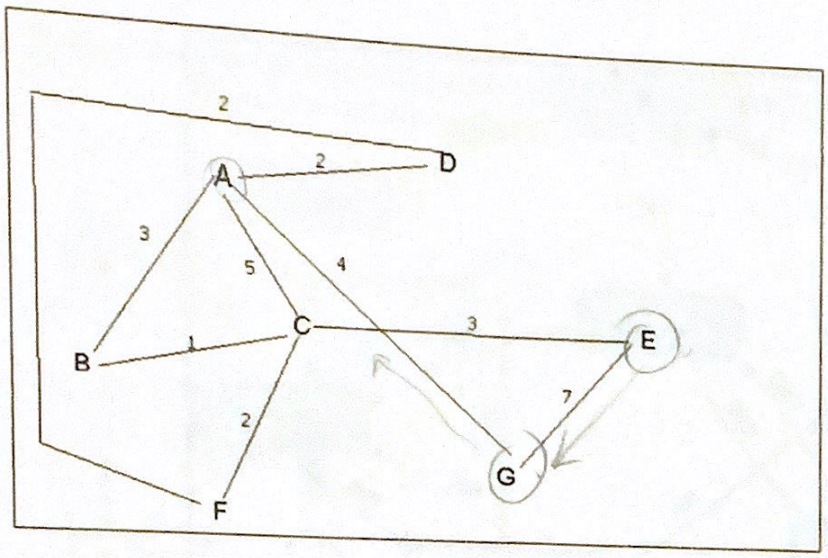
States

Alaska
California
Hawaii
New York
Oregan
Texas
Vermont

Alaska	California	Hawaii	New York	Oregon	Texas	Vermont
0	0	0	0	1	0	0
0	0	0	0	0	0	0
1	1	0	1	0	1	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	0	0	0	1
1	1	0	0	0	0	0

3. b. Show the adjacency lists that would describe the edges in the graph





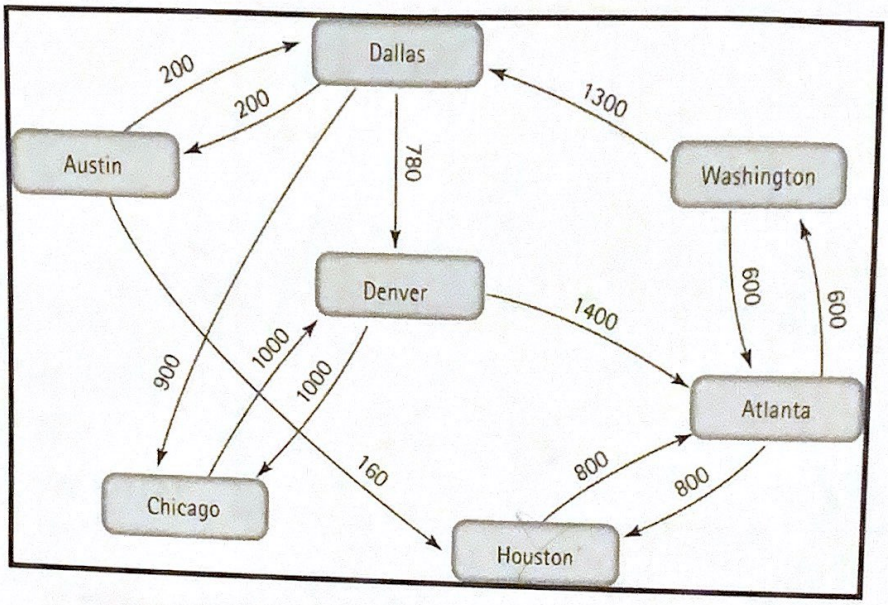
A B C D E F G
A 0 1 1 1 0 0 1
B
C
D
E
F
G

4 a. Which of the following lists the graph nodes in depth first order beginning with E?

A) E, G, F, C, D, B, A
B) G, A, E, C, B, F, D
C) E, G, A, D, F, C, B
D) E, C, F, B, A, D, G

4 b. Which of the following lists the graph nodes in breadth first order beginning at F?

A) F, C, D, A, B, E, G
B) F, D, C, A, B, C, G
C) F, C, D, B, G, A, E
D) a, b, and c are all breadth first traversals



5. Find the shortest <u>distance</u> from Atlanta to every other city

① Atlanta $\xrightarrow[800]{}$ Houston ⇒ 800
direct

② Atlanta $\xrightarrow[600]{}$ Washington ⇒ 600
direct

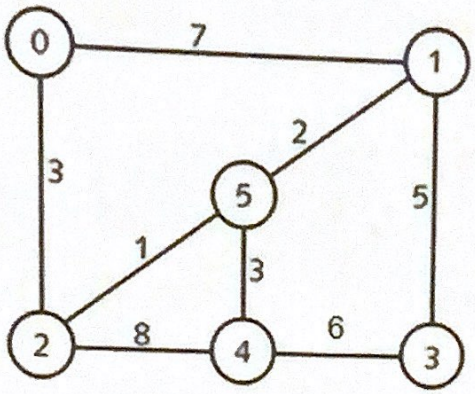
③ Atlanta $\xrightarrow[600]{}$ Washington $\xrightarrow[1300]{}$ Dallas $\xrightarrow[900]{}$ Chicago ⇒ 2800

④ Atlanta $\xrightarrow[600]{}$ Washington $\xrightarrow[1300]{}$ Dallas $\xrightarrow[780]{}$ Denver ⇒ 2680

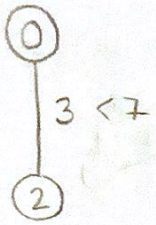
⑤ Atlanta $\xrightarrow[600]{}$ Washington $\xrightarrow[1300]{}$ Dallas ⇒ 1900

⑥ Atlanta $\xrightarrow[600]{}$ Washington $\xrightarrow[1300]{}$ Dallas $\xrightarrow[200]{}$ Austin ⇒ 2100

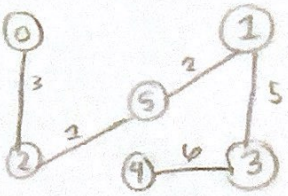
6. Find the minimal spanning tree using Prim's algorithm. Use 0 as the source vertex . Show the steps.



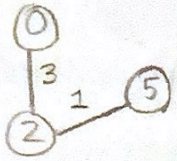
1



5

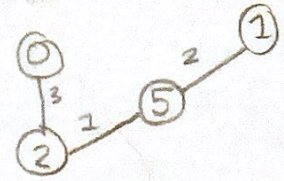


2



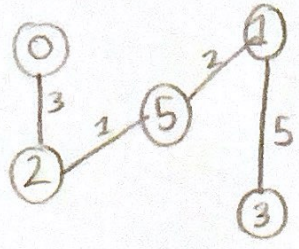
1 < 8

3



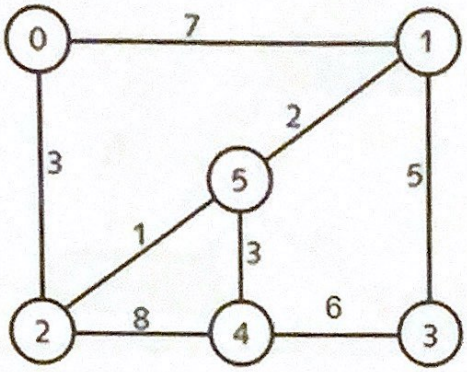
2 < 3

4

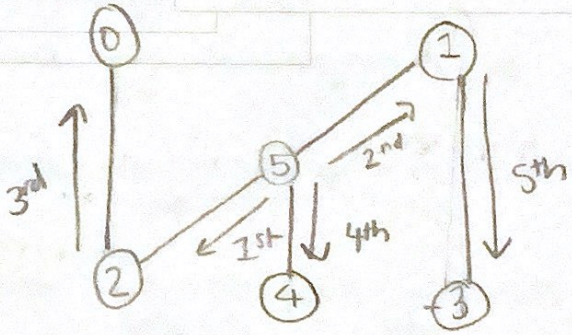


5 < 7

7. Find the minimal spanning tree using Kruskal's algorithm. Show the weights in order and the steps.

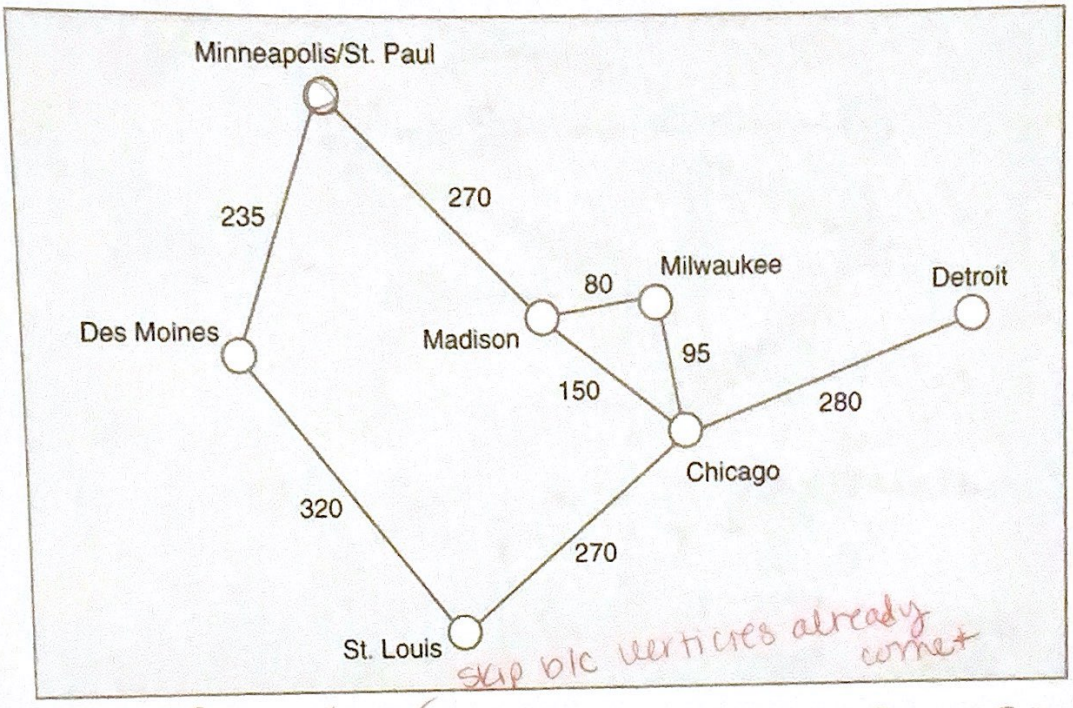


Sorted list
1, 2, 3, 3, 5, 6, 7, 8



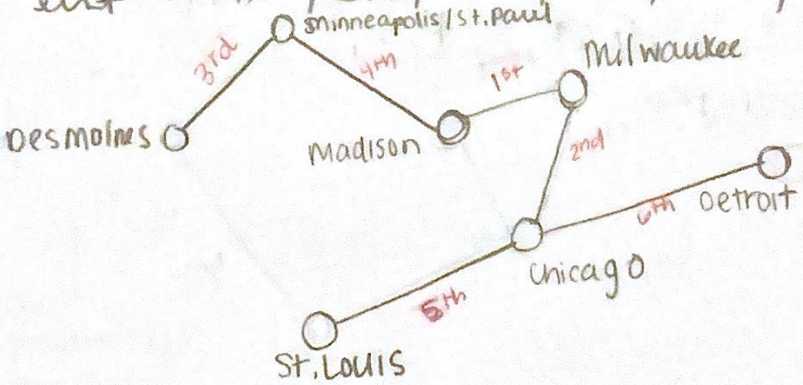
minimal Kruskal's algorithm

8. Find the minimal spanning tree using the algorithm you prefer. Use Minneapolis/St. Paul as the source vertex



skip b/c vertices already connect

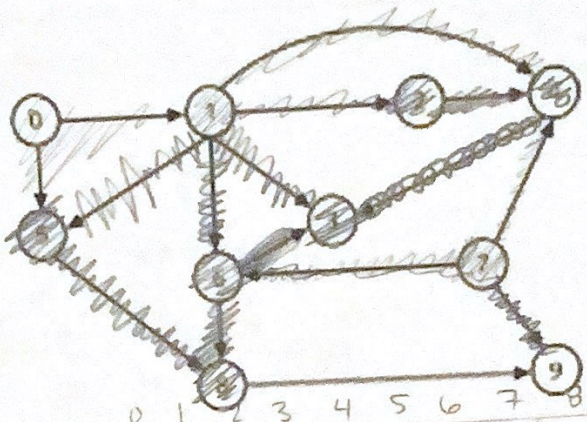
sorted list {80, ~~95~~, 150, 235, 270, 270, 280, 320}



St.Paul-Mad	270	(4)
St.Paul-Desmoi	235	(6)
Des Moins-StL	320	(8)
StL-Chi	270	(5)
Chi-De	280	(7)
Chi-Mil	95	(2)
→ Chi-Mad	150	(3)
SKIP Mil-Mad	80	(1)

V={ Madison, Milwaukee, Chicago, Minn/St Paul, desmoines, St. Louis, Chicago, Detroit }

9. List the nodes of the graph in a breadth first topological ordering. Show the steps using arrays predCount, topologicalOrder and a queue



predCount[i]

0	1	2	3	4	5	6	7	8	9
0	1	4	3	3	2	2	0	2	2
0	0	3	2	2	1	1	0	1	1
0	0	2	2	2	1	1	0	1	1
0	0	1	2	2	1	1	0	1	1
0	0	1	2	2	1	1	0	1	1
0	0	1	2	2	1	1	0	1	1
0	0	1	2	2	1	1	0	1	1
0	0	1	2	2	1	1	0	1	1
0	0	1	2	2	1	1	0	1	1
0	0	1	2	2	1	1	0	1	1

Topological order

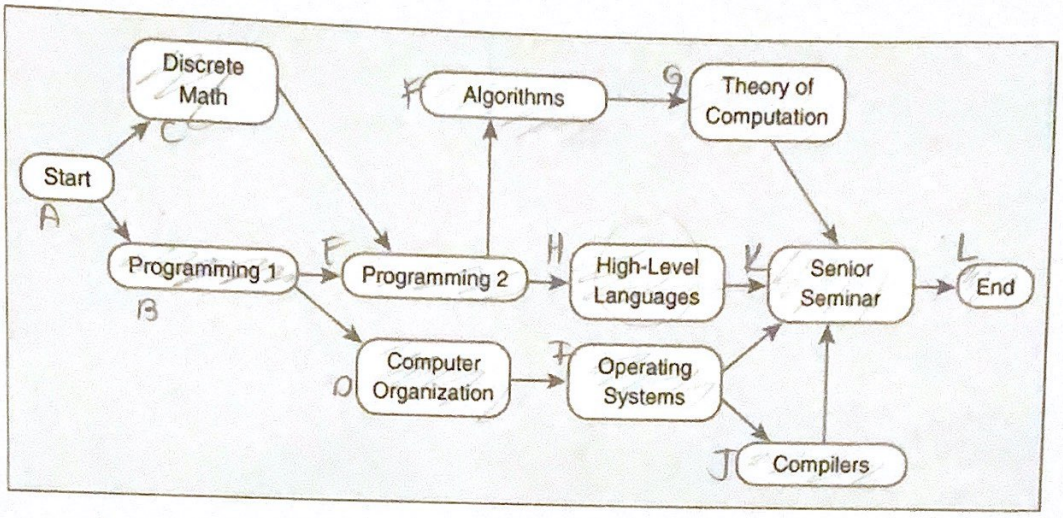
0	7	1	2	4	5	6	3	8	9
0	1	2	3	4	5	6	2	8	9

queue v_K

0	7	1	2	4	5	6	3	8	9

predcount[K] = 0

10. List the nodes of the graph in a breadth first topological ordering.



A B C D E F G H I J K L
0 1 1 1 2 1 1 1 1 1 3 1
0 0 0 1 0 0 0 0 1 0 2 1
0 1 0
3 0

A, B, C, d, e, F, g, h, i, J, K, L

A, B, C, d, e, F, g, h, i, J, K, L

Start
programming 1
discrete Math
computer organization
programming 2
Algorithms
Theory of computations
High level languages
operating systems
compilers
Senior seminar
END